Extraction d'information de sous-catégorisation à partir des tables du LADL

Claire Gardent, Bruno Guillaume, Guy Perrier, Ingrid Falk

TALN 2006, Leuven, 10–13 avril 2006

Extraction d'information de sous-catégorisation à partir des tables du LADL*

Claire Gardent[1] Bruno Guillaume[2] Guy Perrier[3] Ingrid Falk[4]
(1) CNRS/LORIA
(2) INRIA/LORIA
(3) Université Nancy 2/LORIA
(4) CNRS/ATILF
Nancy, France
Prénom.Nom@loria.fr

Résumé Les tables du LADL (Laboratoire d'Automatique Documentaire et Linguistique) contiennent des données électroniques extensives sur les propriétés morphosyntaxiques et syntaxiques des foncteurs syntaxiques du français (verbes, noms, adjectifs). Ces données, dont on sait qu'elles sont nécessaires pour le bon fonctionnement des systèmes de traitement automatique des langues, ne sont cependant que peu utilisées par les systèmes actuels. Dans cet article, nous identifions les raisons de cette lacune et nous proposons une méthode de conversion des tables vers un format mieux approprié au traitement automatique des langues.

Abstract Maurice Gross' grammar lexicon contains rich and exhaustive information about the morphosyntactic and syntactic properties of French syntactic functors (verbs, adjectives, nouns). Yet its use within natural language processing systems is hampered both by its non standard encoding and by a structure that is partly implicit and partly underspecified. In this paper, we present a method for translating this information into a format more amenable for use by NLP systems, we discuss the results obtained so far, we compare our approach with related work and we identify the possible further uses that can be made of the reformatted information.

Mots-clefs : Lexique-grammaire, M. Gross, sous-catégorisation

Keywords: Grammar Lexicon, M. Gross, Subcategorisation

* Nous tenons à remercier Eric Laporte et l'Institut d'électronique et d'informatique Gaspard-Monge d'avoir rendu disponible une version électronique de ces tables. Nous remercions également le projet LexSynt et le Contrat Plan État Région : Ingénierie des Langues, du Document et de l'Information Scientifique, Technique et Culturelle pour le financement partiel des résultats présentés dans cet article.

1 Introduction

De nombreux travaux ont montré que le fait d'avoir des informations de sous-catégorisation détaillées était un élément essentiel pour améliorer la couverture et la précision des systèmes de traitement automatique de la langue. Par exemple, (Briscoe & Carroll, 1993) montre que la moitié des échecs d'analyse sur des données nouvelles est due aux données de sous-catégorisation. Dans (Carroll & Fang, 2004), le taux d'analyses correctes augmente de 15% lorsque la grammaire HPSG (Head Driven Phrase Structure) est enrichie par des informations de sous-catégorisation détaillées.

Pour le français, il existe actuellement deux lexiques de sous catégorisation à large couverture : Proton[1] et LEFFF[2]. Le premier décrit 3700 verbes et 8 600 entrées (un verbe ayant généralement plusieurs entrées selon ses sens possibles) mais n'est pas librement disponible. En outre, comme pour les tables du LADL, le format de Proton est tel qu'il n'est pas directement utilisable par les applications TAL. Le second, LEFFF, contient 207 436 formes fléchies de 5 381 verbes et est disponible sous licence LGPL-LR. Si son format le rend directement compatible avec les applications TAL et en particulier avec une utilisation par des analyseurs syntaxiques, sa production semi-automatique laisse cependant ouverte la question de sa précision et de sa couverture.

Par ailleurs, le lexique-grammaire de Maurice Gross et sa forme électronique dérivée, les tables du LADL, donnent une description systématique des foncteurs syntaxiques du français ; il est partiellement disponible en version électronique sous licence LGPL-LR[3]. Comme le remarque (Hathout & Namer, 1998; Gardent *et al.*, 2005), cette ressource contient des informations de sous-catégorisation qui sont à la fois détaillées et extensives. Ainsi chaque usage d'un lemme est associé à une description de l'ensemble de ses cadres de sous-catégorisation, et pour un cadre donné, chaque argument est associé à une description de ses propriétés morphosyntaxiques et grammaticales. De plus, le lexique-grammaire couvre non seulement les verbes (environ 6 000 lemmes) mais également les adjectifs et les noms (environ 25 000 constructions à verbe support avec tête nominale ou adjectivale).

Afin de pouvoir tirer profit de l'information riche contenue par les tables du LADL, il est d'abord nécessaire de traduire cette information dans un format compatible avec les travaux existants en particulier avec LEFFF. Une fois cette conversion faite, il devient possible de comparer, de fusionner et de compléter une ressource par une autre. On pourra par exemple compléter l'information verbale contenue dans LEFFF par l'information adjectivale contenue dans les tables, ou encore valider par un processus de comparaison et de fusion, les lexiques verbaux contenues par chacune de ces ressources.

Dans cet article, nous proposons une *méthode de conversion des tables du LADL* vers un format approprié pour les applications du TAL. La section 2 présente la structure et le contenu des tables du LADL. Dans la section 3, nous montrons en quoi le format des tables est problématique pour une utilisation dans le domaine du TAL puis nous présentons la procédure de conversion développée. La section 3.4 identifie différentes façons de modifier le contenu du lexique syntaxique produit (SynLex-LADL) et montre comment ces techniques ont été utilisées pour la création du lexique SynLex-TAL. La section 4 compare notre approche avec celle de

[1] `http://bach.arts.kuleuven.be/PA/proton.html`

[2] `http://www.labri.fr/perso/clement/lefff/`.

[3] cf. `http://infolingu.univ-mlv.fr/DonneesLinguistiques/Lexiques-Grammaires/Presentation.html`

Table 8 Description de la table

N0 =: Nhum	N0 =: Nnr	N0 =: le fait Qu P	N0 = V1 W	[extrap]		8	N0 est V-ant	N0 V	N0 est Vpp W	N1 =: Qu P	N1 =: Qu Psubj	[pc z.]	N1 =: si P ou si P	N1 = V0 W	Tc =: futur	Tc =: passé	Vc =: devoir	Vc =: pouvoir	Vc =: savoir	N1 =: ce(ci+la)	N1 =: Ppv	de N1 = de là	N1 =: Nhum	N1 =: N-hum	N1 =: le fait Qu P	Prép Nhum = Ppv	[extrap][passif]	de N1 V N0	N0 V contre Nhum
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
+	-	-	-	-	s'	abstenir	+	+	-	-	+	-	-	+	-	-	-	-	-	+	+	-	-	+	-	+	-	-	-
+	-	-	-	-		abuser	-	+	-	+	+	-	-	+	+	-	-	+	+	+	+	-	+	+	+	+	+	-	-

(Hathout & Namer, 1998). Enfin la section 5 présente les premiers résultats obtenus et résume les axes de recherche à poursuivre pour finaliser la tâche entamée.

2 Les tables du LADL

Contenu. Le lexique-grammaire est organisé en un ensemble de tables, chaque table regroupant les usages des mots prédicatifs qui partagent les propriétés dites définitoires de la table. En particulier, toutes les entrées d'une table ont en commun un (parfois deux) cadre(s) de sous-catégorisation de base. Pour chaque lemme d'une table, les colonnes indiquent en outre des propriétés de sous-catégorisation additionnelles pour ce lemme et en particulier des informations sur :
– les réalisations possibles des arguments (catégorie, préposition, complémenteur, etc.) ;
– les propriétés syntaxiques du verbe ou de ses arguments (réflexivisation, clitisisation, etc.) ;
– les sous-catégorisations alternatives ;
– les possibilités de redistributions (passif long, passif court, etc.).
Pour illustration, la figure 2 donne les deux premières lignes de la table 8. Cette table décrit les verbes avec un complément (nominal ou phrastique) introduit par la préposition "de" (e.g., *Jean se repent de sa conduite*). Les 6 premières colonnes décrivent les propriétés du sujet ; les colonnes 6 et 7, les propriétés du verbes ; les colonnes 8, 9, 10, 29 et 30 donnent les cadres alternatifs ; la colonne 28 indique la possibilité d'avoir un passif impersonnel ; et les colonnes 11 à 27 décrivent les réalisations possibles et les propriétés syntaxiques de l'argument introduit par la préposition "de".

Structure. Les colonnes des tables du LADL sont reliées entre elles par des relations de conjonction, de disjonction et de dépendance. Ainsi, pour la table 8 par exemple :
– les colonnes 13 et 14 *dépendent* des colonnes 11 et 12 (la possibilité d'avoir comme complément une complétive sans préposition ou une proposition interrogative dépend de la possibilité d'avoir un complément phrastique avec le mode indicatif ou subjonctif).
– les colonnes 16 et 17 donnent une information *disjonctive* sur les valeurs de traits atomiques (le complément à l'infinitif est compatible avec un adverbe indiquant le futur, avec un adverbe indiquant le passé, avec les deux ou avec aucun des deux).
– la colonne 2 fournit une information *disjonctive* sur la réalisation de l'argument. (le sujet est "non restreint", c'est-à-dire qu'il peut être un sujet humain, une infinitive ou une complétive).
– les colonnes 6 et 7 fournissent une information *conjonctive* sur le verbe (d'une part le lemme et d'autre part la possibilité d'avoir une particule réflexive "*se*" ou "*s'*").

3 Des tables du LADL à un lexique de sous-catégorisation

Bien que les tables contiennent une information très riche, leur format ne permet pas un usage facile dans les applications de TAL pour plusieurs raisons.

Premièrement, le format n'est pas celui d'un lexique syntaxique TAL. Il est donc nécessaire de convertir les tables en un tel lexique, c'est-à-dire en un lexique qui associe à chaque item prédicatif l'ensemble de ses cadres de sous-catégorisation, et où chacun de ces cadres se présente sous la forme d'un ensemble de structures de traits décrivant le verbe et les arguments permis par le cadre considéré.

Deuxièmement, la structure de la table est souvent implicite, voire absente. Ainsi dans la table 8, la disjonction de valeurs de traits atomiques entre les colonnes 18 à 20 doit être inférée à partir du fait qu'elles partagent le même trait (Vc) dans leur entête. La dépendance entre les colonnes 16 à 21 et la colonne 15 n'est quant à elle indiquée en aucune façon et doit être reconstruite manuellement.

Troisièmement, les entêtes de colonnes nécessitent d'être explicitées dans des structures de traits compatibles (noms de trait, noms de valeur) avec celles utilisées par les systèmes de TAL actuels.

Pour pallier cet obstacle, nous proposons une méthode de conversion des tables qui permet de traduire leur contenu en un format mieux adapté à la création d'une ressource utilisable par les outils TAL. Cette méthode procède en trois étapes :

1. Pour chaque table, un **graphe SynLex** (produit manuellement) reflète l'interprétation qui en a été faite à partir de la table elle-même, de sa description, des livres de l'équipe de Maurice Gross (Gross, 1975; Boons *et al.*, 1976; Guillet & Leclère, 1992) et d'une concertation avec les membres du projet LexSynt (`http://lexsynt.inria.fr/index.php`) dont en particulier, Eric Laporte. Ce graphe rend la structure d'une table explicite et traduit les entêtes en structures de traits.
2. Un algorithme basé sur le parcours du graphe SynLex produit pour chaque entrée (i.e. chaque ligne) d'une table l'ensemble des cadres de sous-catégorisation associé par la table à cette entrée. Le lexique ainsi produit est appelé **lexique SynLex-LADL** et a pour but de refléter au plus près le contenu des tables.
3. Un processus de post-traitement produit à partir du lexique SynLex-LADL un **lexique SynLex-TAL** adapté à une utilisation en TAL.

3.1 Le format du lexique SynLex-LADL

Le lexique SynLex-LADL présente le contenu des tables (modulo les erreurs ou omission qui peuvent résulter du processus de conversion) dans un format standard pour les lexiques syntaxiques c'est-à-dire, un format associant à chaque usage de lemme un cadre de sous-catégorisation exprimant le nombre et la nature syntaxique de ses compléments ainsi qu'éventuellement des informations morphosyntaxiques ou transformationnelles. Ce format est celui adopté par exemple dans le lexique syntaxique ComLex pour l'anglais (Macleod *et al.*, 1994) ainsi que dans les lexiques ACQUILEX (Sanfilippo, 1993), EUROTRA (ten Hacken *et al.*, 1991) et GENELEX (Asstril *et al.*, 1993).

Une entrée dans le lexique SynLex-LADL associe un (usage de) verbe à un cadre syntaxique représenté par une liste de structures de traits désignées par les noms `v`, `a0`, `a1`, `a2` ou `lf`. La figure 1 donne une vue partielle de 6 entrées d'un tel lexique pour les verbes *abuser* et *bénéficier*.

– La structure de traits `v` décrit des propriétés morpho-syntaxiques ou sémantiques du verbe (auxiliaire, mode, particule, . . .).

```
--abuser--
a0[cat=n, type_sem=humain] v[...] lf[passif_impersonnel=vrai]
a0[cat=n, type_sem=humain] v[...] a1[...] lf[passif_impersonnel=vrai]

--bénéficier--
a0[cat=n, type_sem=non_humain] v[...] a1[...] lf[extrap_sujet=vrai, passif_impersonnel=vrai]
a0[cat=p, mode=inf] v[...] a1[...] lf[extrap_sujet=vrai, passif_impersonnel=vrai]
a0[cat=p, mode=ind|subj, comp=le_fait_que] v[...] a1[...] lf[extrap_sujet=vrai, passif_impersonnel=vrai]
a0[cat=p, mode=ind|subj, comp=que] v[...] a1[...] lf[extrap_sujet=vrai, passif_impersonnel=vrai]
```

FIG. 1 – Vue partielle d'entrées du lexique SynLex-LADL

– Chaque structure de traits `a`i décrit un argument du verbe (catégorie, préposition, complémenteur, ...).
– La structure de traits `lf` est différente des précédentes en ce sens qu'elle exprime des possibilités de redistribution des arguments suffisamment régulières pour qu'il ne soit pas nécessaire de les décrire par des entrées lexicales explicites (passivation, extraposition du sujet, passif impersonnel, ...). Ainsi contrairement aux transformations non régulières qui introduisent des cadres dans le lexique de façon explicite (définition extensionnelle), les transformations régulières tels le passif sont indiquées dans le lexique (pour tenir compte des exceptions lexicales) mais n'introduisent pas de cadres supplémentaires (définition intensionnelle).

L'ordre des éléments `v`, `a0`, `a1`, `a2` dans la liste exprime l'ordre linéaire des arguments et du verbe dans une configuration canonique du cadre syntaxique associé.

3.2 Les graphes SynLex

Un graphe SynLex rend explicite la structure d'une table et traduit les entêtes en structures de traits. Par exemple, la figure 2 montre une partie du graphe associé à la table 8[4]. Un graphe SynLex est un graphe acyclique *et/ou* contenant trois types de nœuds : les nœuds **OU**, les nœuds **ET** et les nœuds **CADRE**.

– Les nœuds **OU** indiquent une disjonction d'informations entre leurs nœuds fils. Ils sont représentés par des ovales.
– Les nœuds **ET** indiquent une conjonction d'informations. Ils sont représentés par des rectangles séparés en deux parties : la partie supérieure contient une condition et la partie inférieure contient un ensemble (éventuellement vide) de spécifications de traits. Les conditions sont de la forme :
  – `[`c`]` qui est `True` si la colonne de numéro c contient `+` ou est non vide (dans le cas des colonnes pour le lemme, les prépositions ou les particules) et `False` sinon.
  – `[ !`c`]` qui est `True` si la colonne de numéro c contient `-` ou est vide et `False` sinon.

  Une spécification de trait est de la forme *arg.feat* = *value* où *arg* peut être `v`, `a0`, `a1`, `a2` ou `lf`. Une valeur de trait *value* est soit une disjonction de valeurs atomiques soit le symbole $\$c$ qui indique que la valeur du trait est le contenu de la case $[l, c]$ de la table (l est le numéro de la ligne de la table considérée pour un lemme).
– Enfin, les nœuds **CADRE** décrivent les cadres syntaxiques associés à une table. Ils sont représentés par des rectangles grisés n'ayant pas d'arc entrant. Leur contenu est en deux parties : la condition de la partie supérieure peut être la constante `True` (cas des cadres de base de la table) ou une condition comme celle définie pour les nœuds **ET** ; la partie inférieure indique l'ordre linéaire dans lequel le verbe avec ses arguments composent le cadre syntaxique.

[4]Les expressions entre guillemets sur la figure sont des commentaires et sont ignorés par l'algorithme.

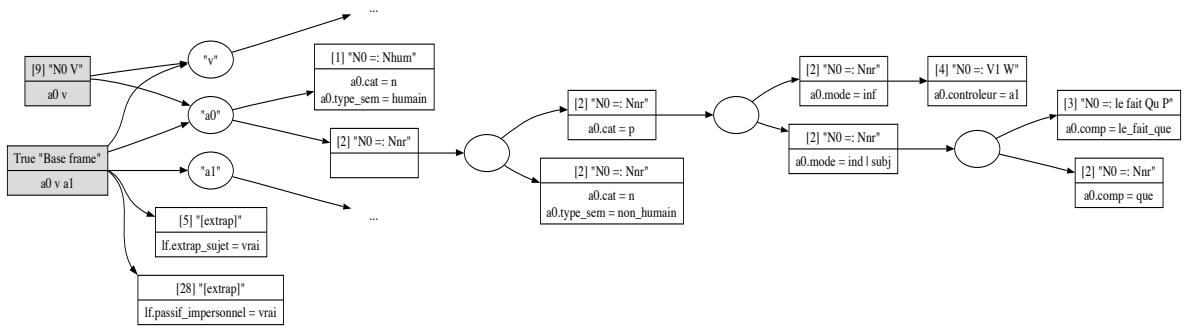

FIG. 2 – Un sous-graphe (simplifié) du graphe pour la table 8

3.3 Traitement du graphe SynLex

Dans une première passe, l'algorithme de traitement du graphe calcule pour chaque ligne de la table un graphe réduit qui reflète l'information associée par la table à cette ligne (i.e., un usage de verbe). Dans ce graphe réduit, chaque aiguillage (i.e. choix d'un arc sortant pour chaque nœud **OU**) donne une entrée dans le lexique SynLex-LADL.

Graphe réduit. À partir du graphe d'une table et d'une ligne l de cette table, on calcule le graphe réduit de la façon suivante :

– pour chaque nœud **ET** et **CADRE** où la condition est `False`, on supprime le nœud et les arcs incidents ;
– pour chaque nœud **OU** sans arc sortant, on supprime ce nœud et ses arcs entrants ;
– on remplace le symbole $\$c$ par le contenu de la case $[l, c]$ de la table.

Dans les nœuds restants, les conditions sont nécessairement `True`, on peut donc les enlever.

La cohérence entre les colonnes de la table se traduit pas le fait que les graphes réduits obtenus doivent être connexes. Par exemple, la colonne 3 dépend de la colonne 2 dans la table 8 ; dans le graphe correspondant, chaque nœud **ET** contrôlé par la condition `[3]` est dans un sous-graphe d'un nœud **ET** contrôlé par la condition `[2]`. Une ligne incohérente dans la table 8 (e.g., colonne 2 avec – et colonne 3 avec +) conduirait à un graphe réduit non-connexe. Lors du traitement de la table, si on obtient un graphe non-connexe, cela signifie soit que le graphe complet n'est pas correct pour cette table, soit qu'il s'agit d'une erreur dans la table.

La figure 3 donne les graphes réduits pour les verbes *abuser* et *bénéficier* de la table 8. On ne donne en fait que la partie de ces graphes correspondants à la simplification du graphe complet présenté dans la figure 2. Dans le tableau ci-dessous, on trouve les cases utiles pour le calcul de ces graphes réduits.

	1	2	3	4	5	9	28
	N0= : Nhum	N0= : Nnr	N0= : le fait Qu P	N0= : V1W	[extrap]	N0 V	[extrap] [passif]
abuser	+	–	–	–	–	+	+
bénéficier	–	+	+	–	+	–	+

Calcul du lexique SynLex-LADL. À partir du graphe réduit correspondant à chaque ligne de la table, l'algorithme calcule les entrées du lexique SynLex-LADL associées par énumération des aiguillages du graphe. Plus précisément :

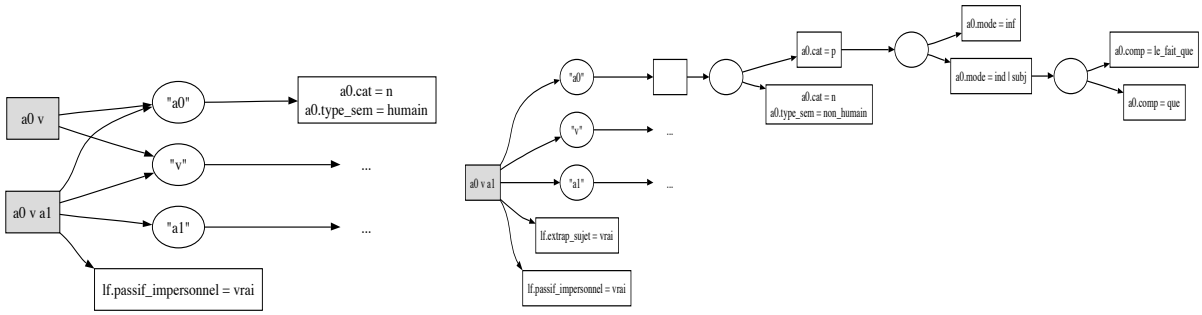

FIG. 3 – graphes réduits pour les verbes *abuser* et *bénéficier*

– pour chaque nœud **CADRE** F, on initialise un cadre syntaxique sous forme d'une liste $[a_{i1} = \emptyset, a_{i2} = \emptyset, \ldots, a_{ip} = \emptyset, v = \emptyset, a_{j1} = \emptyset, a_{j2} = \emptyset, \ldots, a_{jq} = \emptyset, \mathrm{lf} = \emptyset]$ où $\emptyset$ est une structure de traits vide. Cette initialisation est effectuée à partir de la suite $a_{i1}\ a_{i2} \ldots a_{ip}\ v\ a_{j1}\ a_{j2} \ldots a_{jq}$ indiquant dans la partie inférieure du nœud F l'ordre linéaire entre le verbe et ses arguments.
– dans le sous-graphe issu de F, on énumère les aiguillages (un aiguillage est un choix d'une arête sortante de chaque nœud **OU**) et pour chaque aiguillage, on ajoute au cadre initial les traits des nœuds **ET** restants.

Pour nos deux exemples de verbes, on obtient les entrées lexicales de la figure 1.

3.4 Production du lexique SynLex-TAL

Le lexique produit à partir du graphe SynLex reflète le contenu des tables du LADL tel qu'il résulte du processus de conversion décrit ci-dessus. Or l'information contenue dans ces tables peut être incorrecte, incomplète ou superflue pour une application TAL et le processus de conversion peut avoir introduit des erreurs. Par ailleurs, le lexique produit ne présente aucune factorisation de l'information (un usage de verbe a autant d'entrées qu'il a de cadres syntaxiques). Pour pallier ces inconvénients, il est possible de :
– *modifier le graphe SynLex*. Une modification du graphe se répercute immédiatement et de façon systématique dans le lexique produit. La structure du graphe permet en outre de faire ces modifications de façon factorisée (une information ajoutée ou modifiée dans les niveaux supérieurs du graphe aura une portée plus générale qu'une information ajoutée au niveau de ses feuilles).
– *modifier le lexique produit*. Cela est utile dans les cas où les modifications à apporter sont difficiles à inclure dans le graphe de la table ; ou encore dans les cas de filtrage où seule une partie de l'information produite est pertinente.
– *modifier le processus de traitement du graphe*. C'est en particulier le cas envisagé pour produire une version factorisée du lexique.

Pour produire le lexique SynLex-TAL, nous mettons en œuvre chacune de ces techniques de la façon suivante.

Suppression d'information. Certaines informations contenues dans le lexique SynLex-LADL sont très détaillées et ne sont généralement pas utilisées par les analyseurs ou les générateurs. Par exemple, dans la table 8, les colonnes 16 à 20 encodent des contraintes sur les propriétés temporelles et verbales du complément infinitif. Les colonnes 16 et 17 indiquent si l'infinitive peut être combinée avec un adverbial temporel futur ou passé tandis que les colonnes 18 à 20 indiquent si le verbe de l'infinitif peut être *devoir, pouvoir* ou *savoir*. Si ces propriétés sont utiles pour établir des classifications de verbes, elles le sont moins pour l'analyse ou la génération au-

tomatique de textes. En vue d'un usage par les outils TAL, nous produisons un lexique simplifié (SynLex-TAL) qui contient seulement un sous-ensemble des traits utilisés dans SynLex-LADL.

Ajout d'information. Dans les tables du LADL, l'information sur les fonctions grammaticales et les rôles thématiques remplis par chacun des arguments est soit absente (fonction grammaticale) soit implicite et partielle (rôle thématique). Pour inclure cette information dans les lexiques produits, nous avons enrichi les graphes SynLex avec l'information pertinente si bien que les lexiques SynLex-LADL et SynLex-TAL contiennent une information qui n'est pas présente dans les tables d'origine.

Correction d'information. L'information obtenue n'est pas toujours correcte. Les corrections peuvent se faire soit au niveau du graphe, soit au niveau du lexique produit. Par exemple, le graphe de la table 5 permettait de générer des cadres où apparaissaient à la fois le sujet extraposé avec fonction "sujet" et rôle thématique "arg0" et le sujet impersonnel avec fonction "sujet" et rôle thématique "arg0". Il s'agit dans ce cas, d'une erreur dans la définition du graphe qui peut être rapidement corrigée. Il suffit de modifier le graphe en associant au nœud du sujet extraposé la fonction "objet" et en éliminant le rôle "arg0" du nœud associé avec le sujet impersonnel. Un deuxième cas, plus délicat à traiter, est celui des contrôleurs d'infinitive dans la table 4. Dans cette table, le sujet peut être une infinitive dont le contrôleur est l'argument ayant pour rôle thématique "arg1". Or cette information est présente dans le niveau inférieur du graphe. Pour bloquer un cadre intransitif où le sujet demande un contrôleur (arg1) n'existant pas, il faudrait dupliquer dans le graphe toute l'information décrivant les sujets possibles, omettre dans cette duplication le sujet infinitif et relier le cadre n0V à ce nouveau sous-graphe. Une solution moins correcte linguistiquement mais plus rapide est de filtrer dans le lexique les entrées demandant un contrôleur absent du cadre considéré[5].

Factorisation d'information. Pour factoriser le lexique produit nous envisageons de modifier l'algorithme de traitement du graphe. Le graphe est en effet une représentation factorisée des cadres de sous catégorisation possibles (l'information commune est "repoussée" vers la partie supérieure du graphe). Un algorithme de traitement plus sophistiqué devrait permettre de factoriser les lexiques produits.

4 Comparaison avec d'autres travaux

(Hathout & Namer, 1998) propose également une méthode d'extraction à partir des tables du LADL. Celle-ci permet de produire trois formats de lexique : un lexique indépendant d'une théorie particulière, un lexique TAG et un lexique HPSG.

La différence principale entre les deux approches réside dans la façon de coder l'information structurelle (donnée entre autres par la documentation des tables). Nous codons cette information dans des graphes à partir desquels les lexiques sont calculés ; alors que dans (Hathout & Namer, 1998), l'information est insérée directement dans les tables et les entêtes sont traduits automatiquement dans des spécifications de trait. Compte tenu de la complexité des tables et de la difficulté à évaluer les lexiques produits, cette différence n'est pas négligeable. En effet, les graphes SynLex fournissent une spécification déclarative de la structure et du contenu des tables qui facilite leur appréhension. De plus, parce qu'il admet une interprétation procédurale, le graphe SynLex permet de modifier rapidement et de façon consistante les lexiques générés (cf. supra).

[5] Pour l'instant cette correction n'a pas été intégrée.

Une autre différence concerne le traitement des entêtes de colonnes. Alors que dans (Hathout & Namer, 1998) ces entêtes sont traduites semi-automatiquement, nous utilisons une traduction manuelle. Nous avons choisi cette traduction manuelle pour deux raisons. Premièrement, certaines entêtes sont en fait traduites par un sous-graphe (et pas seulement par un nœud). C'est le cas par exemple pour l'entête "N0 = Nnr" qui indique un sujet non-restreint et qui se traduit dans nos graphes par un sous-graphe disjonctif qui définit le sujet comme un groupe nominal non-humain ou une proposition (infinitive ou non). Deuxièmement, certaines entêtes doivent être interprétées en fonction du contexte ; elles ne sont pas traduites de la même façon dans toutes les tables où elles apparaissent. Par exemple, l'entête "N1 = QuPsubj" indique un complément phrastique avec un mode subjonctif dans la table 10, alors que dans la table 8, la même entête indique un complément phrastique avec un mode subjonctif et introduit par la préposition "de" (car, par défaut, dans cette table, N1 est introduit par "de").

5 Résultats et perspectives

La méthode présentée ci-dessus a été appliquée aux tables disponibles compilées par l'équipe de Maurice Gross[6]. Pour chacune de ces tables, nous avons défini un graphe SynLex et produit les lexiques SynLex-LADL et SynLex-TAL correspondants. Ces lexiques sont disponibles sur le site SynLex[7]. Notons que ces lexiques sont des résultats préliminaires dont la finition passe par un travail à venir important de validation et de complétion. Leur mise à disposition vise uniquement à faciliter les retours d'information.

Le travail actuel et futur vise les objectifs suivants : évaluer la qualité des résultats obtenus, étendre l'approche à l'ensemble des tables disponibles et structurer SynLex-TAL sur la base de familles verbales à la Levin.

Evaluation. Il importe d'évaluer la qualité des lexiques produits et leur utilité pour le TAL. Outre une évaluation "manuelle" des résultats obtenus, nous poursuivons dans cette optique plusieurs axes de recherche en parallèle.

Un premier axe consiste à comparer le lexique produit avec un lexique existant. Nous travaillons en particulier avec les auteurs de LEFFF en vue d'unifier les formats (traits et structure) utilisés par SynLex et LEFFF. Une fois ce travail accompli, il deviendra possible de comparer voire de fusionner les deux ressources. Pour chaque entrée, il sera en particulier possible de mesurer les recouvrements et les différences, les recouvrements permettant de valider les cadres concernés et les différences de corriger soit des carences (un cadre manquant dans l'un des lexiques), soit des erreurs.

Un deuxième axe consiste à utiliser analyseurs et corpus afin de détecter par les échecs à l'analyse les cadres manquants et par les analyses où l'ambiguïté est trop grande, les cadres superflus ou redondants.

Enfin, on pourra également utiliser la génération. Pour un cadre de sous-catégorisation sur lequel plusieurs ressources diffèrent, les phrases générées à partir de ce cadre donneront des informations sur sa validité.

Extension. La méthodologie mise en place permet de traiter d'autres tables relativement rapi-

[6]Ces tables décrivent les verbes qui ont un complément phrastique. Maurice Gross a construit 19 tables qui couvrent environ 2 500 verbes. Parmi ces tables, 12 sont disponibles sous licence LGPL-LR, elles couvrent 1 936 verbes et 2 019 usages.

[7]`http://www.loria.fr/~gardent/ladl/content/resultats.php`

dement : il “suffit” pour ce faire de créer les graphes représentant chacune de ces tables. L’algorithme de traitement décrit en section 3 permet ensuite de produire pour chacun des graphes créés les lexiques SynLex-LADL et SynLex-TAL correspondants. Néanmoins les tables utilisées jusqu’ici font toutes partie de celles qui ont été directement supervisées par Maurice Gross. Il importe de voir dans quelle mesure l’approche SynLex se généralise aux autres tables traitées par le LADL.

Structuration. Comme le montre (Levin, 1993), les verbes peuvent être organisés en classes syntaxiques qui mettent en évidence des similarités sémantiques entre les verbes. Le lexique-grammaire de Maurice Gross va dans ce sens, les tables regroupent les usages de verbes en fonction de leur propriétés syntaxiques. A long terme, l’objectif est la création à partir des ressources dérivées des tables, d’un VerbNet pour le français similaire à celui créé pour l’anglais par (Kipper *et al.*, 2000).

Références

ASSTRIL, GSI-ERLI, FRANCE I. & GROUP S. (1993). *Rapport sur la couche syntaxique*. Rapport interne, Projet Eureka Genelex.

BOONS J.-P., GUILLET A. & LECLÈRE C. (1976). *La structure des phrases simples en français. I : Constructions intransitives*. Droz, Genève.

BRISCOE E. & CARROLL J. (1993). Generalised probabilistic LR parsing for unification-based grammars. *Computational Linguistics*.

CARROLL J. & FANG A. (2004). The automatic acquisition of verb subcategorisations and their impact on the performance of an hpsg parser. In *Proceedings of the 1st International Joint Conference on Natural Language Processing (IJCNLP)*, p. 107–114, Sanya City, China.

GARDENT C., GUILLAUME B., PERRIER G. & FALK I. (2005). Extracting subcategorisation information from maurice gross’ grammar lexicon. *Archives of Control Sciences*, **15**(LI), 253–264.

GROSS M. (1975). *Méthodes en syntaxe*. Hermann.

GUILLET A. & LECLÈRE C. (1992). *La structure des phrases simples en français. Constructions transitives locatives*. Droz, Gen ève.

HATHOUT N. & NAMER F. (1998). Automatic construction and validation of french large lexical resources : Reuse of verb theoretical linguistic descriptions. In *First International Conference on Language Resources and Evaluation, Granada, Spain*, p. 627–636.

KIPPER K., DANG H. T. & PALMER M. (2000). Class based construction of a verb lexicon. In *Proceedings of AAAI-2000 Seventeenth National Conference on Artificial Intelligence*, Austin TX.

LEVIN B. (1993). *English verb classes and alternations : a preliminary investigation*. Chicago University Press.

MACLEOD C., GRISHMAN R. & MEYERS A. (1994). COMLEX syntax : Building a computational lexicon. In *Proceedings of COLING ’94*, p. 268–272.

SANFILIPPO A. (1993). Lkb encoding of lexical knowledge. In *Default Inheritance in Unification-Based Approaches to the Lexicon*. Cambridge : CUP.

TEN HACKEN P., MAAS H. & MAEGAARD B. (1991). Dictionaries in eurotra. In *The Eurotra Linguistic Specifications*. The Commission of the European Communities.